

Advanced Automation in Space Shuttle Mission Control

Troy A. Heindel
National Aeronautics and Space Administration
Johnson Space Center

Arthur N. Rasmussen
MITRE Corporation

Robert Z. McFarland
UNISYS Corporation

Abstract

The Mission Control Center (MCC) at NASA's Johnson Space Center in Houston is certainly one of America's foremost technological achievements. From the early days of Apollo through Skylab to the Space Shuttle program, Mission Control has played an integral part in our ability to send humans into space and return them safely. Up until three years ago the technology of the MCC had remained virtually unchanged; flight controllers were supported by minimal tools and were expected through ponderous amounts of diligence and training to monitor the health of the country's leading aerospace products. The Real Time Data System (RTDS) Project was undertaken in 1987 to introduce new concepts and technologies for advanced automation into the MCC environment. The project's emphasis is on producing advanced near-operational prototype systems that are developed using a rapid, interactive method and are used by flight controllers during actual Shuttle missions. In most cases the prototype applications have been of such quality and utility that they have been converted to production status. A key ingredient has been an integrated team of software engineers and flight controllers working together to quickly evolve the demonstration systems.

Background

The Mission Control Center (MCC) has been the heart of NASA manned space flight operations since the Apollo program. It currently actively supports the Space Shuttle missions and will provide support for upcoming manned missions such as Space Station *Freedom* as well. The MCC is organized as a hierarchy of flight control officers headed by the flight director and organized into "disciplines" each of which monitors a specific portion of the Shuttle's onboard systems. The flight director is the leader of the flight control team and bears final responsibility for all mission decisions. Each discipline consists of a sub-team of controllers headed by a "front room" controller who supports the flight director and who in turn is supported by the "back room" controllers for the discipline. The organization of the MCC is shown in Figure 1.

In the past, the Mission Control Center (MCC) has relied exclusively on mainframe computers to process and display spacecraft data on monochrome display screens located in the flight control consoles. Although state of the art at the time of their installation, the systems have aged and now lag considerably behind current technologies. This is most evident in these systems' user interface which are clearly "user-unfriendly" by today's standards. The systems

provide a primarily textual display of raw spacecraft data and require flight controllers to spend as much as 60% of their time converting raw data into the information needed to manage the mission[1]. Because of the low level of automation, a flight controller needs more than just a good understanding of the Shuttle's systems; the controller must spend many hours in simulated missions learning to quickly evaluate the raw data, build mental models that match the situation, evaluate them and come to a decision for the appropriate action. Developing the ability to perform these tasks in real time requires many hours of training and means a controller may spend as much as two or three years before becoming certified to support actual missions.

There are several additional factors that make the use of automated monitoring systems highly desirable in the MCC. NASA has a troublesome bi-modal age distribution in its personnel as shown in Figure 2. Due to a hiring freeze between the Apollo and Shuttle programs, there are two distinct populations of flight controllers consisting of highly-experienced Apollo-era veterans who are near retirement age and "Shuttle-only" flight controllers with less than five years of flight control experience. Although the Shuttle is possibly one of the most thoroughly documented pieces of hardware in the world, there is still a considerable body of uncaptured knowledge which only the Apollo veterans maintain. In each of the sixteen flight control disciplines, there are as few as one or two of these veterans remaining. The impending retirement of these veterans in the near future means the average experience level of most flight control disciplines will therefore diminish substantially.

Another contributing factor is the attrition level. Trained operations personnel are highly desired for new manned programs such as Space Station *Freedom*. Since the Shuttle program is the only source of such people, there is a natural migration of highly trained flight controllers to these new and exciting programs. The resulting high rate of attrition requires that new people be trained on a continuing basis, with the length of training time required further aggravating the situation.

The RTDS project was formed to meet the challenge of these problems. The guiding vision of the project is to demonstrate the use of advanced automation to improve the quality of real time flight decisions and thereby increase flight safety and mission success rates. Important components of this are the capture of knowledge, improvements in shortening training time and increasing its

effectiveness, and containment of the growth of the size of flight control teams. The latter is especially important to providing operational support at affordable cost for long duration missions such as Space Station *Freedom* and manned planetary missions.

System Architecture

The applications within the RTDS project have been developed with three basic goals in mind: capture the knowledge and experience of expert flight controllers, decrease flight controller training time, and reduce the flight control team size. Much work has been done in laboratories on the design and implementation of advanced automation systems but in most cases the work remained unnoticed and isolated in the labs. Early on it was decided that the RTDS project would take the most mature of these technologies and demonstrate their use in the operational setting of the Mission Control Center. It was strongly felt that unless the technologies and techniques could be demonstrated in an actual operational setting, they would continue to encounter high resistance and slow acceptance due to the isolated and unproven nature of the laboratory systems.

This decision required that the RTDS system's architecture be designed for use in the operational setting. Because of the pressing demands of the active schedule of Shuttle flights, there has been a natural reluctance to modify existing operational systems to permit the testing of new technologies. This mandated that the RTDS systems would be independent of the existing mainframe-based flight control consoles and would operate in parallel with them. This parallel approach has yielded several unanticipated benefits. First is improved response time: the RTDS data acquisition system shaves 3 to 4 seconds from the 6 second data latency experienced by the existing mainframe system. Second, the existing system provides an immediately accessible standard against which the accuracy and effectiveness of the RTDS systems can be clearly and independently evaluated.

The need for an independent system produced a requirement for an end-to-end real time data system that could process the Shuttle's telemetry stream and deliver the data to demonstration applications for synthesis into information directly useful for flight control needs. The platform selected for the RTDS applications is a distributed environment comprised of Unix-compatible engineering workstations networked using the TCP/IP protocol. This environment was selected because of its flexibility, standardization and cost-effectiveness.

To support effective processing of real time data in this environment, a four layered architecture was designed. Each layer in the architecture plays a role in refining the data from a raw state into information. The layers are clearly defined and independent so that developmental evolution and testing can be performed in parallel. The architecture is shown in Figure 3.

In the first layer, data retrieved from a commercial telemetry processor travels by direct memory access (DMA) into a ring of raw data buffers maintained in a shared memory of the engineering workstation. Data is then removed from the ring, processed and finally placed into one of four application interface buffers, also resident in shared memory. Application programs in the workstation use library routines to retrieve the data from the interface buffers.

The raw data buffers are filled in rotation from the telemetry processor and are needed because the telemetry processor has extremely limited internal storage. The ring of buffers acts as a "rubber band" between the constant data rate coming from the telemetry processor and the subsequent processing of the data. This design is required to enable an operating system not designed for real time operations to support the continuous acquisition of data; the elasticity of the buffer ring compensates for the system load dependent rate of processor switching.

The telemetry processor performs the majority of decommutation prior to delivering the data to the workstation computer. The data is removed from the ring of buffers and processed to complete the decommutation of the data. The processed data is placed in an application buffer; each application buffer contains the data from one major frame of the telemetry stream and the buffers are used in round-robin rotation. (The Shuttle sends one major frame to the ground each second; each major frame contains a snapshot of the values of all onboard systems and sensors for that second.) Each time the application requests data, the application interface routines determine which is the most current application buffer and deliver data to the application from that buffer. The rotation of the application buffers allows an application to attach to a buffer and extract data from it without concern for the data being immediately overwritten.

The application buffers contain not only the data but also an indication of the "staleness" of the data. Each datum has an associated status that indicates whether that datum was received in the major frame contained in the buffer. This approach has been demonstrated to be much superior to the more common "current value table" (CVT) paradigm (in which the most recent value for each datum is made available without regard to the age of the sample). Although the CVT approach may be adequate for some situations, thorough analysis of shuttle telemetry requires time-homogeneity including the ability to determine how two values are related in time. Many situations cannot be properly analyzed with data values that are not bounded in time.

In addition to providing real-time telemetry data, layer one of the RTDS data acquisition provides a recording and playback facility that allows recording real time data as it is received. This has provided a major advance in capability for the flight controller: prior to RTDS, playback of real time data required the entire MCC facility be configured and operating. The RTDS playback allows flight controllers to review data independently at each workstation and has proven invaluable for several purposes. As an example, a recent launch was "scrubbed" just before liftoff due to a problem in the main engine area. Flight controllers were able to replay the data immediately after the scrub and doing so aided in quickly isolating the problem. This in turn allowed correcting the problem so the launch could be retried the next day, saving several days of extremely costly delay. The playback capability is also proving extremely useful for testing new applications as well as for verification and regression testing.

A tool has been developed to control the playback facility. It was dubbed "VCR" because its graphical interface has been made to closely resemble the remote control from a typical home video cassette recorder. The VCR tool allows playback of the data at varying rates, permits "rewinding" and "fast forwarding" and also allows replay points to be set so that the desired section of a recording can be repeatedly replayed automatically.

An Ethernet™ distribution system has also been developed for RTDS that allows multiple workstations to receive real time data from a source workstation. The source workstation can be obtaining data from either a telemetry processor or from the playback facility. This permits multiple workstations to share a single telemetry processor and provides redundancy since workstations receiving data from one telemetry processor can be reconfigured to receive data through a workstation instead.

The layer one software is written in the "C" language. It has been ported to several of the popular engineering workstations and additional porting is currently being performed.

The second layer of the architecture provides generic data manipulation which does not require domain-specific knowledge. This includes conversion of machine dependent floating point formats and calibration of raw data (PCM counts) into engineering units. This layer is also implemented using the "C" language.

The third layer supports domain-specific algorithms. This includes limit checking and calculations based on multiple parameter values. As part of the RTDS project a tool for building algorithm building tool called "CODE" (computation development environment). This tool allows non-programmers such as flight controllers to develop algorithms using a very high level, graphically-oriented language. CODE then translates the high level language into "C" code and links the algorithm to the real time data acquisition and workstation communication facilities within RTDS.

The fourth layer employs rule-based techniques to support both algorithmic and heuristic knowledge. Because of the real-time nature of RTDS, this layer is called upon only when third layer algorithms detect significant changes in the data values. A commercial off-the-shelf real time expert system shell, G2™ from Gensym Corporation, is used to implement the rules as well as an object-oriented graphical user interface.

All four layers communicate with each other and the flight controller through shared memory. The interfaces between the layers are designed to provide a high degree of visibility into the operation of the layers. This is important not only to facilitate testing but more importantly to provide the flight controller with the ability to examine the operations being performed. The latter is proving use for training and is a key ingredient in the acceptance of the RTDS system by experience flight controllers.

Development Philosophy

RTDS is an in-house project. Past experience has shown that direct user involvement is necessary in order to quickly deploy useful systems. For this purpose, the RTDS team is comprised of both development engineers and flight controllers. Several of the expert system applications have been developed primarily by flight controllers with occasional consultation with development support personnel. During the course of the project there has been migration of personnel between the areas resulting in a gain of strength in each.

Past NASA programs were forced in many cases to do ground-breaking engineering in areas such as processing of telemetry data. This approach is still necessary in some areas but can be avoided (at considerable savings in cost and development time) through the use of standardized, commercially available products. The RTDS project has demonstrated such use in several areas. Telemetry processing is done using a commercially available, fully-programmable telemetry processor. The computer hardware and operating system platform is Unix-based with plans for being based on the POSIX standards and new products such as operating systems that are Unix-compatible and provide true real time capability. Network communications are performed using TCP/IP and Ethernet; user interfaces operate under X-windows. The use of these standard products not only saves the cost and time of development, it also makes it possible to easily upgrade components to improve performance and take quick advantage of the cost effectiveness of new technologies.

Although the development strategy is suitable for producing useful applications quickly it does not guarantee that these same systems will be maintainable in the future. We chose to develop and or buy several tools which would ensure a high degree of maintainability. The G2 expert system shell has been extremely useful for this. The RTDS project has developed a set of standards which have been layered on top of G2 so that all applications built using the tool have the same look and feel. Additionally, the G2 tool has many knowledge management facilities which make maintenance an easier task.

Application Areas

The first application area selected was an expert system to support the Integrated Communications Officer (INCO). The INCO flight controllers monitor all communications systems on the Shuttle. As an initial area of investigation, the onboard payload communications system was selected as a system to be monitored by a rule-based expert system[2]. This system was used to monitor the payload communications system during the STS-26 mission, the first flight after the Challenger accident. The system represents several "firsts" in the MCC including the first use of a rule-based expert system and first use of a color graphics-based user interface in Mission Control.

One of the earliest and important RTDS applications created was an application that graphically monitors the Shuttle main engines and analyzes their performance. Just a few months prior to the launch of STS-26, analysis of data from test firings of Shuttle main engines showed flight controllers that certain conditions of main engine performance could lead to key engine valves "locking up". The data needed to diagnose the condition during actual missions was not available from the mainframe system and could not be made available for at least 6 months. As an interim, the controllers decided to read data from the console displays and manually enter the data into a personal computer which would perform the analysis to detect the condition.

The controllers also requested RTDS to examine the problem and propose a solution. Using the RTDS system, project personnel created a functional display containing nearly all the needed data in less than a week. By the time of the STS-26 launch, an application had been developed that performed the desired analysis and produced a graphical display as well. The application was certified for use in support of Shuttle missions and is currently in use during all Shuttle missions.

The RTDS project's Data Communications Officer Expert System (DATACOMM) is the first attempt in the MCC at position automation. Built using the G2 shell, DATACOMM performs all of the data monitoring tasks of the Data Communications Officer. The system currently does not yet send commands to the Shuttle but this is being considered. The data monitoring tasks include tracking data from Shuttle systems that is recorded on the onboard operational recorders as well as monitoring the health and status of related communications equipment. Once complete, DATACOMM will allow the merging of two flight control positions, reducing the INCO team from four persons to three. DATACOMM has been used during shuttle simulations with favorable results and will be tested during the STS-35 mission. To date, four person months have been spent developing DATACOMM. When finished it is estimated that a person-year will have been spent on development and testing.

The Jet-Control Expert System (Jet-Control) was developed for the Guidance, Navigation, and Control (GNC) officer. There are 38 primary Reaction Control System (RCS) jets on the shuttle which provide on-orbit attitude control. In the event that one or more of these jets should fail it is the job of the GNC officer to determine the control capabilities that have been lost. In the past, the GNC officer has used a time consuming twenty-five page paper procedure for making this determination. Jet-Control automatically makes the determination using telemetry data. Additionally, Jet-Control allows the GNC officer to perform "what-if" analyses with the remaining jets to quickly assess the remaining control capabilities and to do in-depth analysis of remaining equipment. During STS-31 (Hubble Space Telescope) Jet-Control detected the failure of three of the RCS jets; the GNC officer used the what-if capability to determine that the shuttle was one jet failure away from a loss of control in the +X direction (forward translation). Jet-Control was built in G2 in four months by one person.

The Remote Manipulator System (RMS), otherwise known as the Shuttle "arm", is vital to the success of missions such as the Hubble Space Telescope deployment. To aid the RMS flight controllers, RTDS personnel have developed a three-view display application for monitoring the position of the arm. Position monitoring is critical to ensuring that the arm is not over-stressed and that neither the arm nor any attached payload can collide with any part of the Shuttle. The application replaces a complicated off-line system that used a separate computer and three display screens and required a flight controller to manually enter each of the arm's multiple joint angles whenever they changed. The RTDS application has proven very useful and is very popular with the RMS controllers.

Visualizing the state of the Shuttle based on telemetry data is a problem faced by many members of the flight control team. Like the RMS arm position, the attitude and movement of the Shuttle is such a problem area. To demonstrate the potential of a graphical approach, an RTDS-based application has been developed that displays the Shuttle's flight instrumentation graphically. The display mimics the Shuttle's attitude and situation instruments and has been described by one astronaut as almost like being in the cockpit. The application is proving very useful for quickly and accurately determine Shuttle attitude and movement during all flight phases. It has also served as a demonstration of the proposed "glass cockpit" retrofit for onboard Shuttle instrumentation.

Of all flight control positions, the flight director is the most difficult. Filling the position requires a thorough knowledge of the Shuttle's systems as well as operational procedures. An RTDS application has been developed to assist the flight director with one of the more difficult tasks of the position, monitoring the weather at the launch site and the multiple possible landing sites around the world. Prior to the RTDS system, the flight director analyzed weather data chiefly by hand, with support from a weather officer. The RTDS application presents the sites on a display that shows the current weather in detail and indicates those data that area out of acceptable limits for ascent or landing. Reaction from flight directors has been very positive and are prompting requests for additional similar capabilities in other areas.

Technology Transfer

Much of the technology that has been developed by RTDS is being used by other data systems projects within NASA. The training division of JSC's Mission Operations Directorate is using the RTDS data playback capability to create standalone flight controller training. The per hour cost of a "full up" shuttle simulation is about \$15,000. With an increasing flight rate it is more and more difficult to schedule enough training time to certify all trainee flight controllers. The stand-alone training capability will not only be cost effective, but will allow NASA to maintain an ample supply of certified flight controllers to meet the busy flight schedule.

NASA's Ames-Dryden Flight Test Facility, located at Edwards Air Force Base in California, employs the RTDS data acquisition system for telemetering the X-29 and F-18 research projects. The X-29 forward swept wing airplane requires timely (at least 100 times a second) monitoring and control of its control surfaces. The F-18 project is exploring the sparsely understood phenomena of "high alpha flight" or high angle of attack. The Air Force Test Flight Research center, also located at Edwards, is using RTDS data acquisition for the F-15 Short Takeoff and Landing (STOL) project in which modified jet engines are being evaluated as short takeoff and enhanced maneuverability options for the F-15.

The shuttle telemetry that is acquired by RTDS is distributed to other users besides flight controllers. RTDS has developed several data distribution methods which include direct memory access, Ethernet, and modem. Real time data can be displayed on office personal computers and is being used to evaluate the "office-based support" concept for the Space Station Control Center project. The data acquisition system of RTDS is being used by the Engineering Directorate of JSC to provide data for IMU testing and data

archiving. The data acquisition system drivers and several of the user tools have been transferred to the Mission Control Center Upgrade (MCCU) project for incorporation into this larger upgrade effort.

Flight controllers of Mission Control have embraced the automation technologies which have been provided to them by RTDS. They have adopted these new tools into their flight controller tool boxes and have as a consequence developed new concepts for monitoring their systems. As past applications have been strongly based on established operations principals, future applications will continue to be. A new facility being incorporated into RTDS is a capability for applications to share information between workstations using network communication. None of the expert systems that have been developed in the MCC currently use this capability; they are stand-alone, isolated applications. This is extremely dissimilar to the actual functioning of the flight controllers who use them. The ability of flight controllers to function as a coordinated team is probably the most important single factor in the successful support of each mission. With the complexity of spacecraft increasing, the notion of team becomes even more important; no one person or machine can understand the system in its entirety. It is for this reason that in the coming months several of the stand-alone expert systems will be linked to each another. This linkage will undoubtedly spawn a whole new class of problems, but these problems must be overcome if we are to realize the full potential of this technology in Mission Control.

References

1. John F. Muratore, Troy A. Heindel, Terri B. Murphy, Arthur N. Rasmussen and Robert Z. McFarland, Space Shuttle Telemetry Monitoring by Expert Systems in Mission Control. In *Innovative Applications of Artificial Intelligence*, H. Schorr and A. Rappaport, Ed., AAAI Press, 1989, pp. 3-14.
2. Arthur N. Rasmussen, John F. Muratore and Troy A. Heindel, The INCO Expert System Project: CLIPS in Shuttle Mission Control. In *Proceedings of the First CLIPS Users Conference* to be published August, 1990.
3. D. A. Mackall, M. D. Pieckett, L. J. Schillin and C. A. Wagner, *The NASA Integrated Test Facility and Its Impact on Flight Research*, NASA Technical Memorandum 100418, Ames-Dryden Flight Research Facility, 1988, 14 pages.

Trademarks

Unix, G2 and Ethernet are trademarks of AT&T Corporation, Gensym Corporation and Xerox Corporation respectively.

Symbols and Abbreviations

COTS	Commercial Off the Shelf
GNC	Guidance, Navigation and Control
INCO	Integrated Communications Officer
JSC	Johnson Space Center
LAN	Local Area Network
MCC	Mission Control Center
MCCU	Mission Control Center Upgrade
MOD	Mission Operations Directorate
MMACS	Mechanical, Manipulator, and Crew Systems
RCS	Reaction Control System
RTDS	Real Time Data System

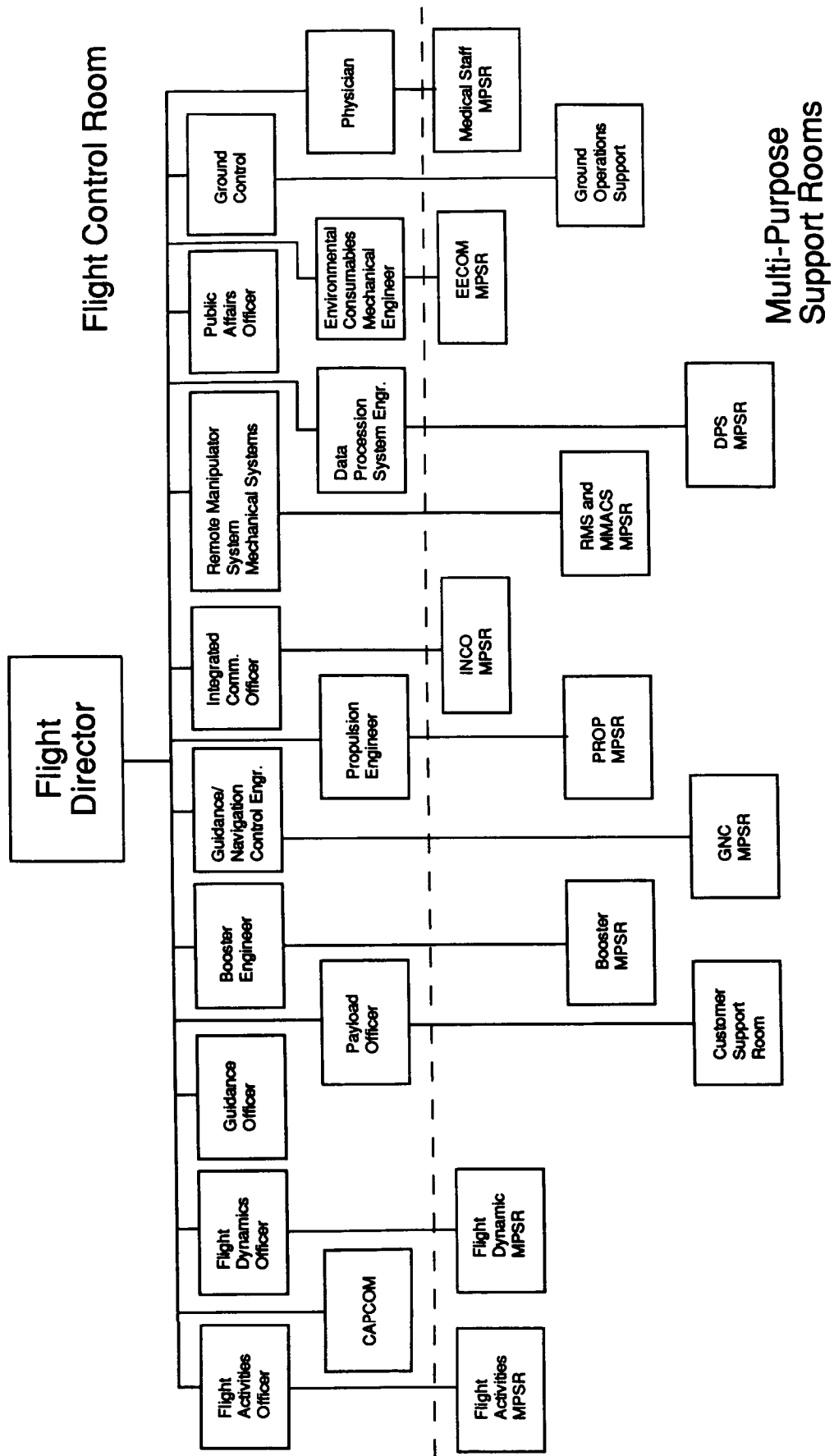


Figure 1. Mission Control Room Organization

S & E Age Profile Comparison

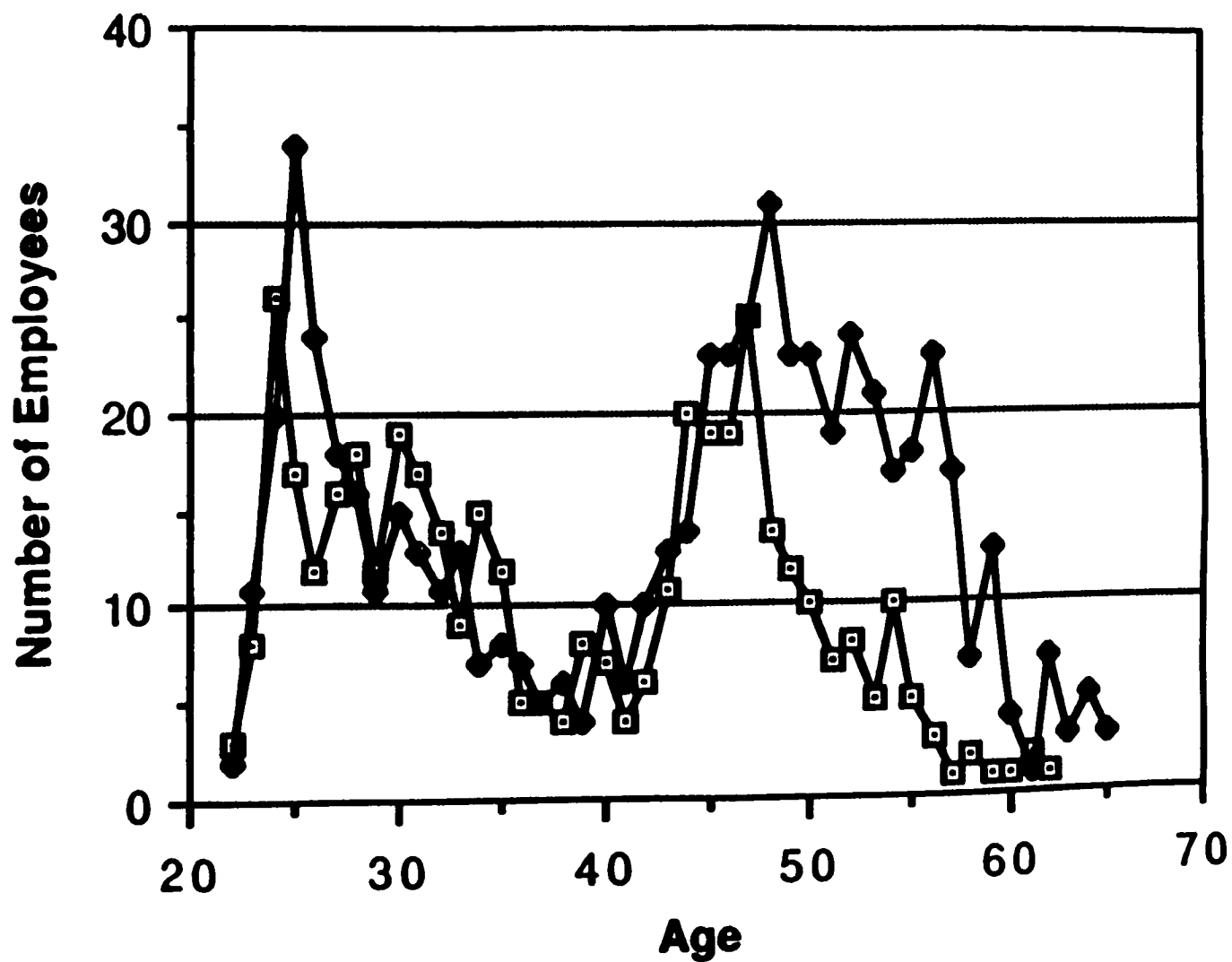


Figure 2. Bi-modal Age Distribution

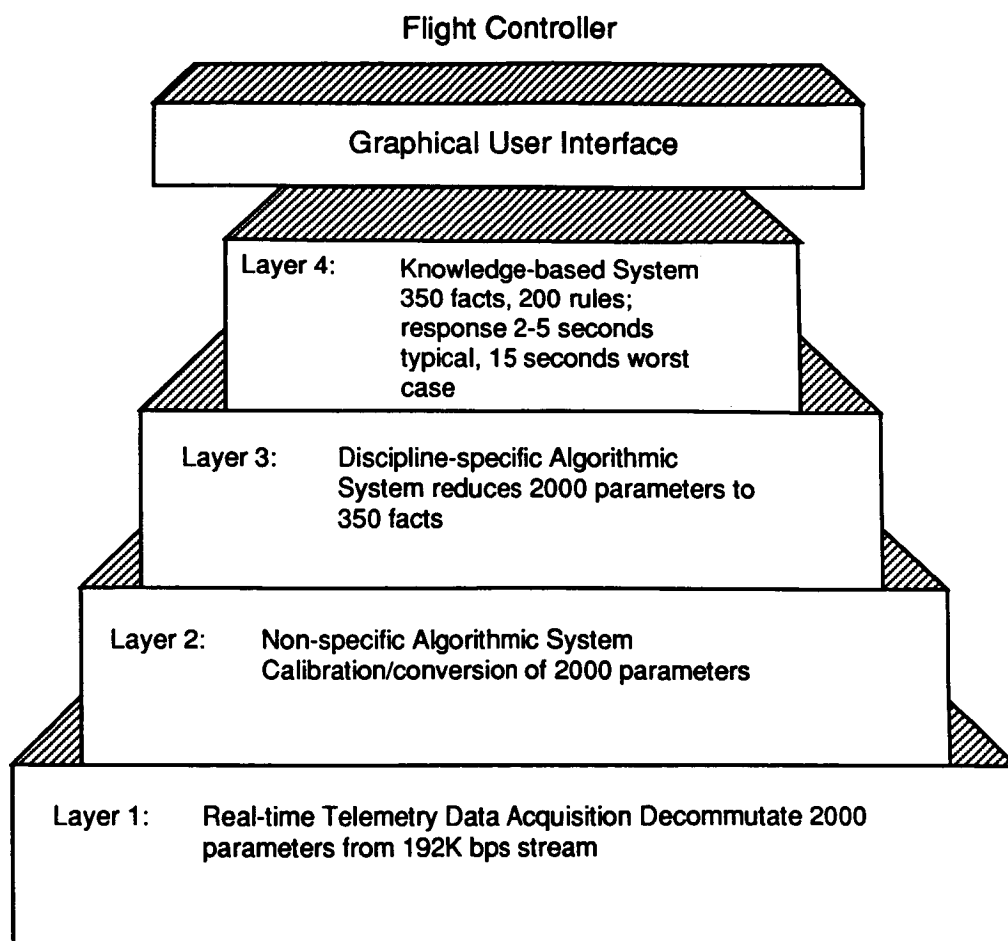


Figure 3. RTDS Four Layer Architecture